AIAA 2001-2978

ADAPTIVE IDENTIFICATION OF FLUID-DYNAMIC SYSTEMS

Aravind Pillarisetti and Louis N. Cattafesta III
University of Florida
Gainesville, FL

31st AIAA Fluid Dynamics Conference & Exhibit
11-14 June 2001 / Anaheim, CA

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **JUN 2001**	2. REPORT TYPE	3. DATES COVERED **00-00-2001 to 00-00-2001**
4. TITLE AND SUBTITLE **Adaptive Identification of Fluid-Dynamic Systems**		5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of Florida,Department of Electrical and Computer Engineering,Gainesville,FL,32611**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S) 11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**12**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

AIAA 2001-2978

ADAPTIVE IDENTIFICATION OF FLUID-DYNAMIC SYSTEMS

A. Pillarisetti† and L. Cattafesta§
Department of Aerospace Engineering, Mechanics, and Engineering Science
University of Florida
Gainesville, Florida 32611-6250
(352) 846-3017, (352) 846-3028 (FAX), catman@aero.ufl.edu

Abstract

Fluid-dynamic systems are inherently nonlinear and are subject to a combination of coherent and random unsteady disturbances. As a result, accurate low-order dynamic models are difficult to obtain for real-time control of such systems. Therefore, controllers implementing adaptive on-line system identification are ideally suited to flow control problems. Adaptive linear and nonlinear filters for real-time system identification are presented in this paper. The linear models studied are traditional FIR and IIR filters, and the nonlinear models include a 2^{nd}-order Volterra filter and the Bilinear filter. The coefficients of the adaptive filter models are calculated and updated using two of the most popular recursive methods, the normalized LMS and RLS algorithms. The adaptive filters are tested offline in software and then implemented on real-time DSP hardware. The focus of this study is on model accuracy and viability in real-time applications. The real-time performance is measured in terms of achievable sampling frequency. Specific applications to relevant nonlinear systems, a spring-mass damper model and a drag-law problem, are also considered in detail.

Introduction

In many flow-control problems; measurable parameters such as pressure, velocity and wall shear stress may vary over time. Relevant applications include active control of flow separation and flow-induced cavity oscillations. In regards to the latter, cavity flows exist in landing gear bays, weapon delivery systems, and in a variety of instrument installation configurations. The control of cavity dynamic loads is an issue of considerable importance to any military aircraft employing internal weapons, examples of which include the B-2, B-1B, F-117, F-22, and JTF. Weapons bay acoustics also impacts stealth technology, which requires minimum radiated noise. Flow-induced cavity oscillations are quite complex. Previous experiments [1,2] have shown significant nonlinear coupling between the dominant modes of oscillation. In particular, time-frequency analysis on unsteady pressure data has revealed time-dependent switching between the primary Rossiter modes. A controller for such oscillations must rapidly and effectively identify this mode-switching phenomenon. Hence, an effective real-time, adaptive identification scheme should be incorporated into the control process.

The problem of adaptive system identification is generally referred to as the determination of a system model by observing its input-output relationship. The input signal is the actuator signal, and the output signal is an appropriate sensor signal. Systems with a single actuator and sensor are referred to as single-input/single-output (SISO) systems. We restrict our attention in this paper to SISO systems.

Adaptive control systems are traditionally applied to systems with quasi-stationary dynamics. However, adaptive control of many fluid-dynamic systems, such as cavity oscillations, requires fast response time to capture non-stationary behavior. This means that the execution of numerical algorithms for control and identification must be completed within the sampling interval. Our working definition of "real-time" computations is those that are completed within one sample interval. The Shannon sampling theorem dictates that the sampling frequency must be at least twice that of the largest frequency component in the measured signal. Since many flow-control experiments are conducted in scaled wind-tunnel facilities, the relevant frequency components increase in direct proportion to the scale

† Graduate Research Assistant. Student Member AIAA.
§ Assistant Professor. Senior Member AIAA.

reduction factor. This, in turn, increases the computational requirements for the real-time control system. As algorithms become more complex, faster processors and parallel processing may be required. Fortunately, the sustained increase in microprocessor computational power and the decrease in their cost permit the possibility of implementing complex identification and control algorithms. In this paper, we focus on adaptive system identification because it is a prerequisite for effective feedback control.

Adaptive system identification is not new, having been applied successfully in diverse fields such as communications, geophysical exploration, industrial robots, vibration, and noise control. However, the applications to flow control have been few [3,4,5]. Rathnasingham and Breuer [3] successfully used a finite impulse response (FIR) filter to identify the linear system dynamics in the near wall region of a turbulent boundary layer. Mouyon et al. [4] developed a physics-based system model to implement a feedback control scheme to maintain a fixed transition location in a boundary layer. Unfortunately, the model failed to capture sufficient information, emphasizing the practical need for on-line identification methods. Finally, Allan et al. [5] used off-line system identification methods with some success to develop a linear discrete state-space model for closed-loop control of separation on an airfoil.

In this paper, as a first step towards adaptive control, we critically examine adaptive system identification using digital filters. Both linear and nonlinear filters are investigated. The purpose here is to test the viability of these filters in a real-time environment that typifies fluid-dynamic applications. Recursive algorithms, namely the Least Mean Square (LMS) and Recursive Least Square (RLS), are used to update the coefficients of the filter model. The paper is outlined as follows. First, the basic theory of linear FIR and infinite impulse response (IIR) filters for system identification are reviewed. Two nonlinear models, based on Volterra and Bilinear series, are addressed next, followed by a description of the LMS and RLS algorithms. Applications to an analytical nonlinear spring-mass-damper model and a quadratic drag-law problem are also considered.

Adaptive System Identification

The first step in adaptive model control is to construct a model for the dynamic system that is used to determine appropriate control inputs to the system that will cause desired system outputs. System identification deals with the problem of building mathematical models of dynamical systems based on observed data from the system. As observed by Ljung [6] the construction of model from data involves three basic entities:

1. Model data from experiment or simulation,
2. Selecting a set of candidate models, and
3. Criterion to evaluate the candidate models.

An adaptive filter can be used to model the behavior of a physical dynamic system. The adaptive filter can be regarded as an unknown "black box" having one or more inputs and one or more outputs. Modeling a SISO dynamic system, as proposed by Widrow and Stearns [7], is illustrated in Fig. 1.

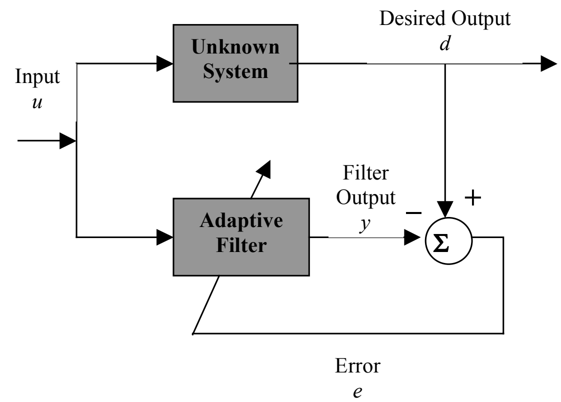


Fig. 1. Modeling of a SISO system using an adaptive filter.

Both the unknown system and the adaptive filter are driven by the same input u. The adaptive filter adjusts itself (using a recursive algorithm) such that its output y matches that of the unknown system d. If the adaptive filter has the correct structure, a perfect fit is possible in a noise-free environment. Note that if the assumed model structure (e.g., the number of poles and zeros) is incorrect, then the resulting parameters of the model may not provide any realistic physical insight. In any case, when the desired and actual input-output responses match, the adaptive filter is said to be a "good" model of the unknown system. Furthermore, if the input is *white noise* (*i.e.*, contains equal power in all frequencies) or its equivalent and if the structure of the adaptive filter enables a good match of the input-output response, then minimizing the mean-square error between d and y will produce the desired adaptive model.

Adaptive Filter - Models

A model of a system is a description of its properties. The most important step in system identification is to determine a class of models within which to conduct a search for a suitable system model. In this section we shall discuss linear and nonlinear filters for this purpose.

Linear Filters

There are two types of linear filters – FIR and IIR models. The FIR filter is the simplest linear model and derives its name from the fact that the duration of its response to an impulse is finite in extent. Its output is a moving average of the input in the sense that the model output is a weighted sum of previous inputs. A FIR filter of length N can be expressed as

$$y(n) = b_0 u(n) + b_1 u(n-1) + \ldots + b_{N-1} u(n-N+1) \quad (1)$$

Here, y is the output, u is the input, and n is discrete time. In polynomial form Eq. (1) becomes

$$y(n) = \mathbf{B}^T(n) \cdot \mathbf{U}(n) = \sum_{i=0}^{N-1} b_i(n) u(n-i), \quad (2)$$

where the superscript T denotes the transpose. $\mathbf{B}(n)$ and $\mathbf{U}(n)$ are given by

$$\begin{aligned} \mathbf{B}(n) &= [b_0(n), b_1(n), \ldots, b_{N-1}(n)]^T \\ \mathbf{U}(n) &= [u(n), u(n-1), \ldots, u(n-N+1)]^T. \end{aligned} \quad (3)$$

$\mathbf{B}(n)$ is the input weight vector, and $\mathbf{U}(n)$ is the input vector, both at time n. The motivation for the FIR model comes from the fact that a linear system output can be expressed as a convolution sum

$$y(n) = \sum_{i=0}^{\infty} g_i u(n-i) \quad (4)$$

where g_i is the impulse response. The FIR model in Eq. (1) is an approximation to the convolution sum in Eq. (4). Since, for stable systems, the coefficients g_i decay to zero as $i \to \infty$, such an approximation is possible. The inherent stability of a FIR model is a distinct advantage when the system to be identified is stable.

The IIR model is an alternative to a FIR model. The model output is a weighted sum of previous inputs as well as previous outputs. An IIR filter model of length N can be expressed as

$$\begin{aligned} y(n) &= a_1 y(n-1) + \ldots + a_N y(n-N) \\ &\quad + b_0 u(n) + \ldots + b_{N-1} u(n-N+1) \end{aligned} \quad (5)$$

or

$$y(n) = \sum_{i=1}^{N} a_i y(n-i) + \sum_{i=0}^{N-1} b_i u(n-i). \quad (6)$$

In polynomial form, this becomes

$$y(n) = \mathbf{A}^T(n) \cdot \mathbf{Y}(n) + \mathbf{B}^T(n) \cdot \mathbf{U}(n), \quad (7)$$

where $\mathbf{A}(n)$ and $\mathbf{Y}(n)$ are given by

$$\begin{aligned} \mathbf{A}(n) &= [a_1(n), a_2(n), \ldots, a_N(n)]^T \\ \mathbf{Y}(n) &= [y(n-1), y(n-2), \ldots, y(n-N)]^T. \end{aligned} \quad (8)$$

$\mathbf{A}(n)$ is the output weight vector, and $\mathbf{Y}(n)$ is the output vector, both at time n. IIR filters use output feedback and have poles and zeros. Consequently, stability is not guaranteed. IIR models are often used to identify systems whose impulse response has a long duration. There have been two fundamental approaches to adaptive IIR filtering that correspond to different formulations of the prediction error; these are known as the equation-error (EE) and output-error (OE) methods. Shynk [8] provides considerable insight into the advantages and disadvantages of the two approaches.

Nonlinear Filters

A simple but highly pervasive type of nonlinearity in fluid dynamics is polynomial in nature (*e.g.*, quadratic or cubic). Unlike the case of linear systems, which are completely characterized by the system impulse response function, it is impossible to find a unified framework for describing nonlinear systems. Consequently researchers working on nonlinear systems are forced to restrict themselves to certain nonlinear models that are less general. Models based on a Volterra series and polynomial nonlinear models are some popular models studied. Two specific cases are considered in detail, a truncated Volterra-series representations and recursive bilinear difference equations, to relate the input-output response of a nonlinear system.

Using the same notation as above, the truncated Volterra series is given by [9]

$$\begin{aligned} y(n) &= \sum_{m_1=0}^{N-1} h_1(n) u(n-m_1) + \ldots \\ &+ \sum_{m_1=0}^{N-1} \ldots \sum_{m_p=0}^{N-1} h_p(m_1, \ldots, m_p) u(n-m_1) \ldots u(n-m_p). \end{aligned} \quad (9)$$

In Eq. (9) h_p is known as the p^{th}-order Volterra kernel of the system. If the representation is truncated at 2nd-order and only a finite support for kernels h_1 and h_2 is provided, then the output can be expressed in filter form as

$$y(n) = \sum_{i=0}^{N-1} a_i u(n-i) + \sum_{i,j=0}^{N-1} b_{i,j} u(n-i) u(n-j) \quad (10)$$

where a_i and $b_{i,j}$ are the time varying linear and quadratic coefficients of the nonlinear filter, respectively. Eq. (10) is called a 2nd-order Volterra filter of length N and is a natural extension of a linear

FIR filter. Due to their complexity, only 2nd-order Volterra filters are considered in this paper.

An alternate polynomial model is a recursive nonlinear difference equation. The simplest of the models in this category is the bilinear model [10], whose input-output relationship is given by

$$y(n) = \sum_{i=1}^{N} a_i y(n-i) + \sum_{i=0}^{N-1}\sum_{j=1}^{N} c_{i,j} u(n-i)\, y(n-j) + \sum_{i=0}^{N-1} b_i u(n-i) \tag{11}$$

where $c_{i,j}$ is the coefficient matrix for the nonlinear cross-term between u and y. A filter based on Eq. (11) is known as a bilinear filter of length N and is a nonlinear extension of an IIR filter, as seen in Eq. (6). Just as linear IIR models can represent many systems with far fewer coefficients than their FIR counterparts, bilinear filters can model many nonlinear systems with fewer coefficients than Volterra series representations [11].

Adaptive Filters – Algorithms

Adaptive filters operate using recursive algorithms. These algorithms do not require any *a priori* information about the system to be identified. The algorithm automatically updates the coefficients of the filter from one time step to the next. Starting from a predetermined set of initial conditions, it converges to the optimum solution in a stationary environment. In a non-stationary environment, the algorithm tracks the system. A wide variety of recursive algorithms have been developed in the literature for the operation of adaptive filters. Most are derived using either a stochastic gradient approach or a least-squares estimation.

Stochastic Gradient – LMS Algorithm

The stochastic gradient approach is based on minimizing a cost function *J*, also referred to as an index of performance. *J* is defined as the mean-squared error between the desired or actual response, *d(n)*, and the filter output *y(n)*

$$J = E\left[\left|e(n)\right|^2\right] \tag{12}$$

Here $E[.]$ is the expectation operator and $e(n) = d(n) - y(n)$ is the error between the desired system output and adaptive filter output.

Wiener and Hopf [12] formulated the theory for the case of a continuous-time linear filter. Haykin [13] extended this theory to a discrete-time filter. The discrete-time equation is also known as the Weiner-Hopf equation. The LMS algorithm is the recursive version of the Weiner-Hopf equation. Table 1 summarizes the algorithm for the simplest linear filter – the FIR filter defined in Eq. (2).

Initialization

Initialize the weight vector by setting

$$\mathbf{B}(0) = 0$$

For each instant of time, compute the weight vector

$$\mathbf{B}(n) = \left[b_0(n), b_1(n), \ldots, b_{N-1}(n)\right]^T$$

input vector

$$\mathbf{U}(n) = \left[u(n), u(n-1), \ldots, u(n-N+1)\right]^T$$

output and error

$$y(n) = \mathbf{B}^T(n)\cdot\mathbf{U}(n)$$
$$e(n) = d(n) - \mathbf{B}^T(n)\cdot\mathbf{U}(n)$$

Update the weight vector

$$\mathbf{B}(n+1) = \mathbf{B}(n) + \mu e(n)\mathbf{U}(n)$$

Note: μ is the learning-rate parameter or the step-size. For convergence, (in a stationary process)

$$0 < \mu < \frac{1}{u_{rms}^2}.$$

Table 1. LMS Algorithm for a FIR Filter.

For faster convergence rates, the step-size μ can be normalized by the power of the input and thus reduce the uncertainty in choosing an appropriate step-size. A robust step-size LMS adaptive algorithm has been proposed in [14]. Feintuch [15] first formulated the LMS algorithm for an IIR model. LMS algorithms for other filter models have been derived and are documented in [16]. Interested readers can refer to Netto et al. [17] and Shynk [8] for detailed discussions of IIR LMS filters. LMS algorithms have also been derived for nonlinear models. Coker [18] has derived the update equations for the 2nd-order Volterra filter, while Mathews [19] presents an algorithm for updating the coefficients of a Bilinear LMS filter.

The LMS filter is very simple to implement and is computationally the least expensive adaptive algorithm. The order of computational complexity (operations per iteration) of LMS linear filters is $\mathcal{O}(N)$, where N is the length of the filter.

Least Squares Estimation – RLS Algorithm

The method of least squares may be viewed as an alternative to Wiener filter theory [13]. The problem is to estimate the unknown parameters of a multiple linear regression model. The computation starts with

known initial conditions and uses the information contained in new data samples to update the old estimates. The cost function to be minimized is expressed as $\xi(n)$. We follow below the notation of Haykin [13],

$$\xi(n) = \sum_{i=1}^{n} \lambda^{n-i} \left| d(i) - y(i) \right|^2; \; 0 < \lambda \le 1. \quad (13)$$

In general, the factor λ ensures that data in the distant past are forgotten in order to afford the possibility of tracking the variations of new data. By minimizing the cost function, we arrive at a set of equations called the "normal equations" of the filter. These complex equations are simplified by using the Matrix Inversion Lemma [13]. This simplified version forms the basis of the RLS algorithm, which is summarized in Table 2 for a FIR filter.

Initialization

Initialize the algorithm by setting

$$\mathbf{B}(0) = 0; P(0) = \delta^{-1} I$$

where δ is a small positive constant, and I is the $N \times N$ identity matrix.

For each instant of time, compute the weight vector

$$\mathbf{B}(n) = \left[b_0(n), b_1(n), \ldots, b_{N-1}(n) \right]^T$$

input vector

$$\mathbf{U}(n) = \left[u(n), u(n-1), \ldots, u(n-N+1) \right]^T$$

gain vector

$$\mathbf{k}(n) = \frac{\lambda^{-1} P(n-1) \mathbf{U}(n)}{1 + \lambda^{-1} \mathbf{U}^T(n) P(n-1) \mathbf{U}(n)}$$

Output and Error

$$y(n) = \mathbf{B}^T(n) \cdot \mathbf{U}(n)$$

$$e(n) = d(n) - \mathbf{B}^T(n) \cdot \mathbf{U}(n)$$

Update

$$\mathbf{B}(n+1) = \mathbf{B}(n) + \mathbf{k}(n) e(n)$$

$$P(n+1) = \lambda^{-1} P(n) - \lambda^{-1} \mathbf{k}(n) \mathbf{U}^T(n) P(n)$$

Table 2. RLS Algorithm for a FIR filter.

The theory of RLS algorithms has been extended to other filter models. Shynk [8] presents the RLS algorithm for IIR filters, while Mathews [19] has developed RLS algorithms for polynomial filters. The rate of convergence of the RLS algorithm is faster than that of the LMS algorithm. However, the number of floating point operations of the RLS algorithm for linear filters is $\mathcal{O}(N^2)$, where N is the length of the filter.

Choice of Adaptive Algorithm

How does one choose an adaptive filter for the task at hand? According to Haykin [13], one or more of the following factors determines the choice of one algorithm vs. another:

1. *Rate of Convergence.* This is defined as the number of iterations required for the algorithm to converge to the *optimum solution*.
2. *Mean-Squared Error.* This is defined as the value of the mean of the squared error.
3. *Stability*. For the filter to be stable, the poles of the filter should lie inside the unit circle in the z-plane throughout the adaptation process.
4. *Robustness*. An adaptive filter should be robust to small disturbances or noise.
5. *Computational requirements*. The issues of concern include the number of floating-point operations required to complete one iteration of the algorithm and the memory required to store the data and program.

Results and Discussion

Real-Time Implementation

Real-time implementation is a key requirement for adaptive identification of any physical system. A practical implementation uses a Digital Signal Processor (DSP) to perform the necessary computations outlined above. Either a mathematical model may be used in a simulation or input/output data from the physical system may be measured with analog-to-digital (A/D) and digital-to-analog (D/A) hardware. This sampling process ideally requires that all computations for system identification and control be completed within one sample interval. Below, we define two relevant parameters used to evaluate the computational requirements of the algorithms discussed above.

1. *Sample Time* T_s: This is defined as the time interval at which data are sampled. To avoid aliasing, the Nyquist criterion states that the data should be sampled at least two times faster than the maximum frequency content in the data.
2. *Turnaround Time* T_A: This is the time required to gather input data and compute the output for one iteration. T_A determines the minimum sampling time (or maximum sampling frequency) of the model.

Clearly, T_A should be less than T_s for real-time identification.

A real-time development environment has been developed by dSPACE using single- and multi-processor systems. These systems have been

integrated with a MATLAB/SIMULINK programming environment, summarized as follows:

1. MATLAB toolboxes are used to formulate an appropriate model. Ultimately, the system is modeled by linear or nonlinear difference equations (as described earlier).
2. The adaptive model is coded using SIMULINK, a graphical programming environment that runs under MATLAB.
3. The application software generates compiled code for the target DSP from the SIMULINK model.

In this paper, we describe this process for a single-processor dSPACE DS1003 card that employs a TMS320C40 DSP processor from Texas Instruments. The processor runs at 60 *MHz* and the board provides a peak floating point performance of 60 *MFLOPS* (*FLOPS* = floating point operations per second).

Using the methods outlined above, the linear and nonlinear filters described above were benchmarked on the DS1003 board by determining T_A for a single iteration. Fig. 2 shows T_A in microseconds as a function of *N*.

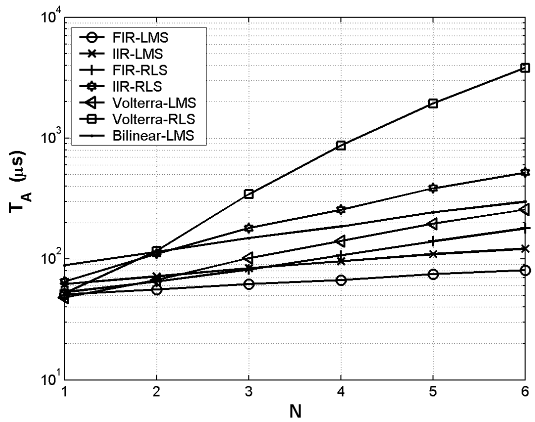


Fig. 2. T_A of adaptive filters on DS1003.

T_A is a linear function of *N* for the LMS-based linear filters. For RLS-based filters, T_A is a nonlinear function of *N*. Furthermore, the nonlinear structure of Volterra and Bilinear filters requires more computations than their linear counterparts. According to Mathews [19], the 2nd-order Volterra LMS algorithm has a computational complexity (operations count per iteration) that is proportional to N^2, whereas the complexity of the RLS algorithm is proportional to N^4. The operation count for each filter type and algorithm is summarized in Table 3. Based on these results, it is evident that implementing nonlinear RLS filters with *N*>3 may not be feasible with the DS1003.

Filter	# of Coeff.	Algorithm	Operation Count
FIR	N	LMS RLS	$\mathcal{O}(N)$ $\mathcal{O}(N^2)$
IIR	$2N$	LMS RLS	$\mathcal{O}(N)$ $\mathcal{O}(N^2)$
Volterra	N^2+N	LMS RLS	$\mathcal{O}(N^2)$ $\mathcal{O}(N^4)$
Bilinear	N^2+2N	LMS RLS	$\mathcal{O}(N^2)$ $\mathcal{O}(N^4)$

Table 3. Dependence of operation count on filter type, algorithm, and length *N*.

T_A can be used to calculate the maximum rate at which data can be sampled. Since T_A is the time required for the DSP to complete one iteration of the algorithm, it is also the minimum possible sample time,

$$T_A \le T_S . \tag{14}$$

The sampling frequency is calculated from T_s by

$$f_S = \frac{1}{T_S} . \tag{15}$$

From Eq. (14) and Eq. (15) we obtain

$$f_S \le \frac{1}{T_A} . \tag{16}$$

Clearly, the maximum sampling frequency is equal to the inverse of T_A.

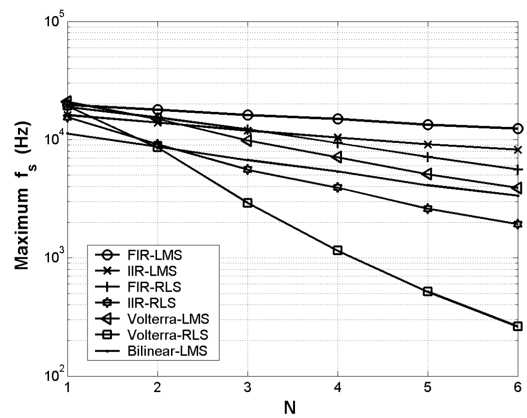


Fig. 3. Maximum achievable sampling frequency for the adaptive filters on the DS1003.

Fig. 3 shows the maximum sampling frequency that can be achieved by the adaptive filters on the DS1003. Fluid-dynamic systems can contain oscillations at high frequencies, particularly in small-scale experiments. For example, previous experiments on active control of cavity oscillations [1,2] showed that the maximum significant frequency ($f_{\max}$) can exceed 1 *kHz*. If an adaptive filter were used to model this system, we would require minimum sampling rates of approximately 2 *kHz*. This effectively eliminates from consideration RLS-based Volterra and IIR filters of length greater than about 3 and 6, respectively. Only LMS-based filters (both linear and nonlinear) will be able to satisfy real-time requirements for this board. Note that much faster boards are currently available. However, keep in mind that a significant portion of the available computational resources will be required by the (as-yet unspecified) adaptive control scheme.

The results of this study may be generalized using a floating-point performance specification of an alternative DSP:

$$f_{s_{\text{target DSP}}} = f_{s_{\text{TMS320C40}}} \cdot \frac{FLOPS_{\text{target DSP}}}{FLOPS_{\text{TMS320C40}}} . \quad (17)$$

Identification of Prototypical Systems

As of this writing, no suitable experimental data were available to evaluate the candidate identification schemes. We therefore opted to study two prototypical nonlinear systems: (1) a spring-mass-damper system – the Duffing Equation and (2) a nonlinear drag-law.

Adaptive filters were used to identify the above two systems. The effectiveness of the adaptive filter in identification was evaluated using the following criteria: (1) accuracy, (2) convergence rate, (3) stability, and (4) robustness to noise.

Duffing Equation

The forced oscillations of a mass attached to a nonlinear spring under the influence of slightly viscous damping exhibits nonlinearities that are typically encountered in fluid-dynamic systems. The equation of motion, known as the Duffing equation [20], has the form

$$m\ddot{y} + c\dot{y} + k_1 y + k_3 y^3 = u . \quad (18)$$

Here, y is the output displacement, u is the input force, and m, c, and k_1 are the mass, damping, and linear spring constants, respectively. The nonlinear spring constant k_3 can be either positive (spring hardening) or negative (spring softening). When $k_3 y^3 \sim k_1 y$ or greater, nonlinearities become significant. This problem is relevant for flow control because many actuators that undergo large deflections can be modeled using Eq. (18).

Using MATLAB and SIMULINK, the Duffing equation described in Eq. (18) was simulated using $m = 1,\ c = 0.5,\ k_1 = 1,\ k_3 = 1$. The input to the system is a single frequency sine wave of amplitude 10 and a frequency of 1.25 *rad/s* (or 0.2 *Hz*), which is close to the natural frequency of 1 *rad/s* for the system linearized about $y = 0$.

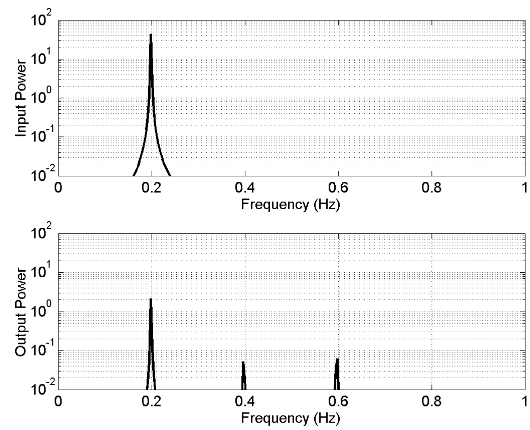


Fig. 4. Power spectrum for Duffing system.

Fig. 4 shows the power spectrum of the input and output of the system. The input spectrum shows a single peak corresponding to the excitation frequency. The output spectrum shows three peaks at 0.2, 0.4, and 0.6 *Hz*. The first peak corresponds to the primary response of the system and the other two are the secondary responses resulting from quadratic and cubic nonlinearities.

LMS-based adaptive filters were employed to identify the above system. Both linear models (FIR and IIR) and nonlinear models (2nd-order Volterra and Bilinear) were explored. Fig. 5 through Fig. 9 show the identified filter output compared to the true system output. Clearly, the nonlinear filters are more accurate than the linear filters, but the linear filters perform surprisingly well. Haykin notes that adaptive filters can be described as nonlinear systems with time-varying parameters [13]. This subtle but significant point is important because a true linear system cannot "generate" harmonics due to its frequency-preserving nature. It appears that adaptive "linear" filters can capture the dynamics of nonlinear systems because of their time-varying nature.

The lengths of the filters were optimized such that they provide the best identification for the least computational expense. The normalized step size was chosen in the range of 0.1-0.5 for the various filters to maximize convergence rates.

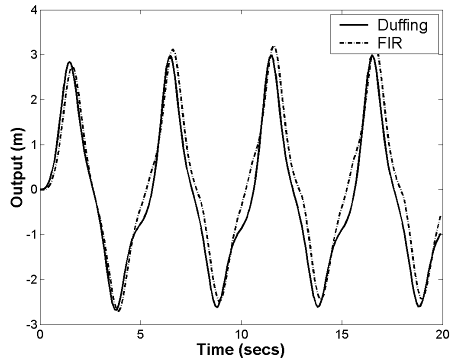


Fig. 5. Duffing ID with a FIR filter (*N*=9).

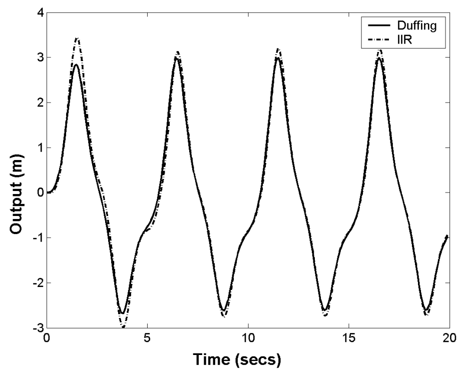


Fig. 6. Duffing ID with an IIR filter (*N*=2).

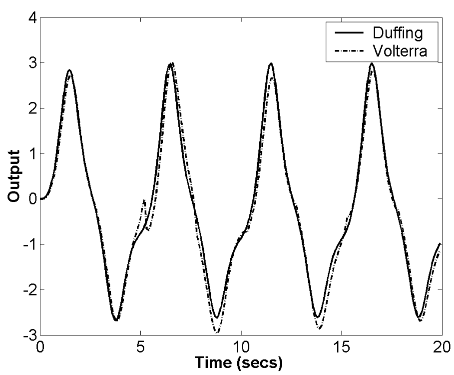


Figure 7. Duffing ID with a Volterra filter (*N*=3).

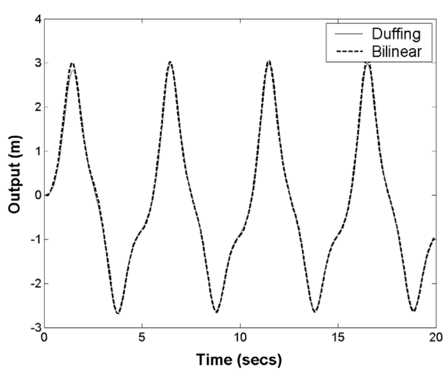
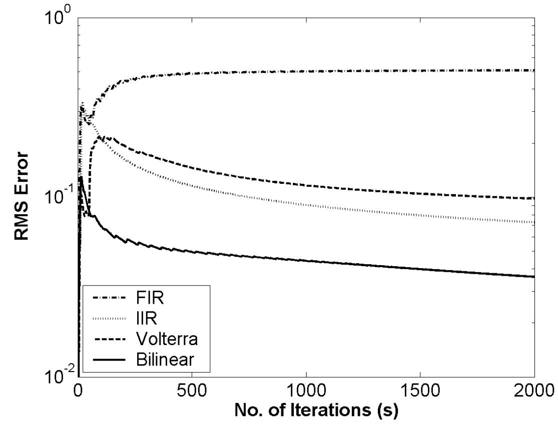
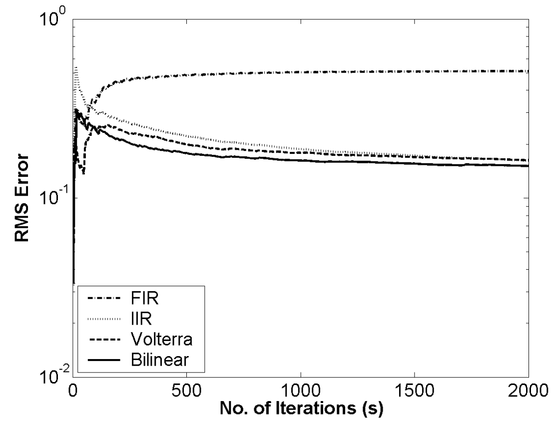


Fig. 8. Duffing ID with a Bilinear filter (*N*=3).

As shown in Fig. 9, the IIR and bilinear filters exhibit better convergence properties than the FIR and Volterra filters. The robustness of the filters to noise is also demonstrated in Fig. 9. Uncorrelated white noise is added to the output of the identified system. The signal-to-noise ratio is 30 *dB* in this case. The filters converge smoothly to their respective optimum solutions. However, the errors are larger than the noise-free case, as expected.



(a)



(b)

Fig. 9. Convergence of adaptive filters for Duffing ID: (a) no noise and (b) noise added to the output.

<u>Nonlinear Drag Law</u>

Another type of nonlinearity that is commonly encountered in fluid dynamics is a nonlinear drag law (NDL) and is a consequence of the unsteady Bernoulli equation along a streamline. Rathnasingham and Breuer used this equation in their synthetic jet model [21]. This system is therefore relevant in flow-control problems.

The governing differential equation is given by

$$\dot{\mathrm{v}} = \frac{\Delta P_0}{\rho L} - \frac{\mathrm{v}|\mathrm{v}|}{2L}\,. \tag{19}$$

Here, v is the velocity, ΔP is the pressure difference along a streamline of length L, and ρ is the fluid density.

The input-output relationship of such a system can thus be written as

$$\dot{y} + ay|y| = bu \; . \tag{20}$$

where y is the output of the system, and u is the input. The constants a and b are given by

$$a = \frac{1}{2L} \text{ and } b = \frac{1}{\rho L} \, . \tag{21}$$

Using MATLAB and SIMULINK, Eq. (20) was simulated, assuming a fluid density of 1.189 kg/m^3 and the streamline length of 2.0 *mm*. The input was a sine wave of frequency 1479 *rad/s*, and the magnitude of the sine wave was chosen such that it produced a maximum velocity of approximately 10 *m/s*. These choices simulate typical experimental conditions [21].

Fig. 10 shows the input and output power spectrum for the above system. The input shows a single peak corresponding to the excitation frequency, and the output shows two peaks, one at the excitation frequency and another at the third harmonic that is characteristic of a cubic nonlinearity.

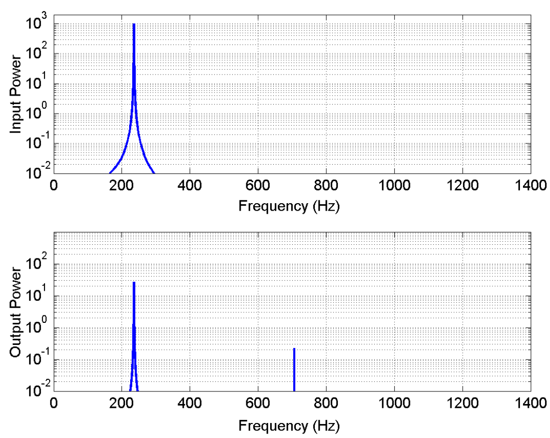


Fig. 10. Spectrum of input and output of the NDL problem.

Only the results for the LMS-based filters are presented here. As in the previous case, N is optimized and the normalized step size is in the range of 0.1-0.5. Fig. 11-Fig. 14 show the identified filter output as compared to the system output. The IIR, 2nd-order Volterra and Bilinear filters closely match the system output, while the FIR filter output is somewhat less accurate.

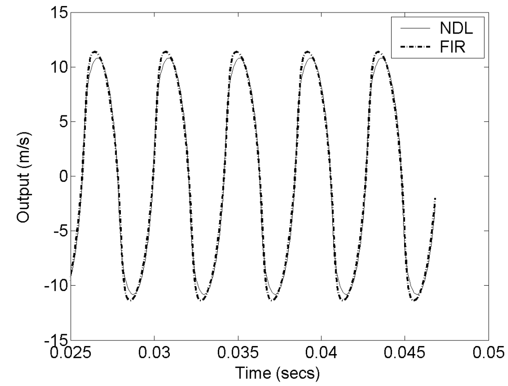


Fig. 11. NDL ID with a FIR filter (*N*=11).

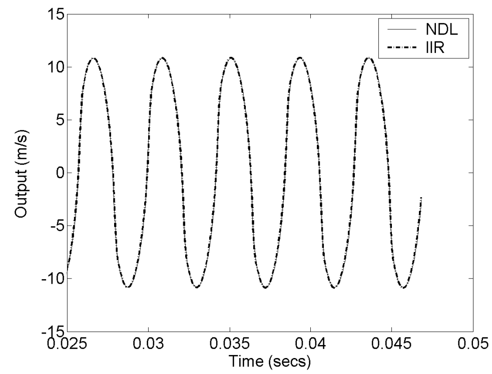


Fig. 12. NDL ID with an IIR filter (*N*=2).

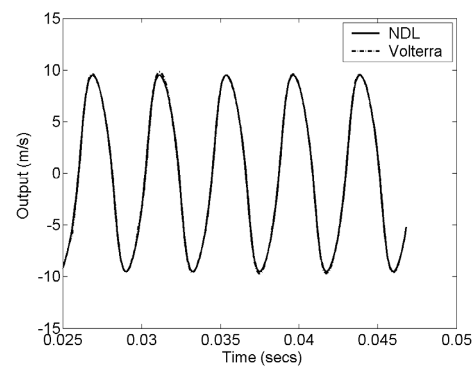


Fig. 13. NDL ID with a Volterra filter (*N*=3).

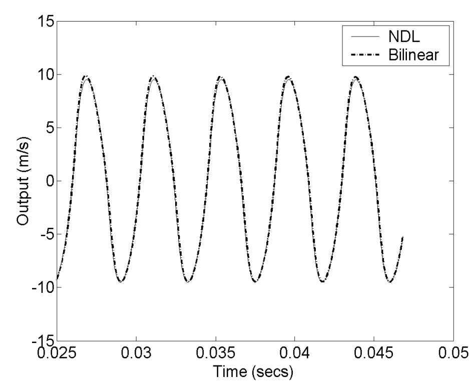


Fig. 14. NDL ID with a Bilinear filter (*N*=3).

The above examples demonstrate that both nonlinear filters and linear IIR filters can be effective in identifying a nonlinear system. However, in real-time control applications, the data are often influenced by noise, in the form of unmodeled dynamics, measurement noise and, in fluid flows, turbulence. Fig. 15 shows the response of the NDL system when uncorrelated white noise is added to the output (SNR = 20 dB). An IIR filter of length 2 is used since a filter of length 1 was apt to be unstable. The presence of noise can lead to biased errors in the identified model when an IIR EE formulation is used [8]. To avoid this bias error and improve robustness, Shynk [8] suggested the alternative OE formulation of the IIR filter. The OE formulation provides unbiased identification but has slower convergence than the EE formulation. Also the OE filter may converge to a local minimum, depending on initial conditions.

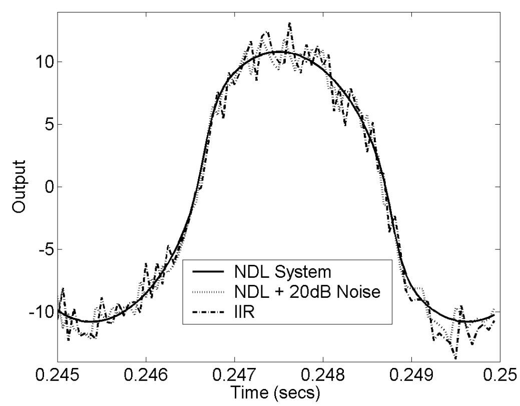


Fig. 15. NDL ID with uncorrelated white noise.

Stability can be maintained by z-domain methods that geometrically enforce stability [22]. In the case of linear IIR filters, the stability of the filters can be determined by calculating the location of the poles. In the discrete-time domain, the poles must lie within the unit circle for stability for all iterations. Fig. 16 shows the locus of the two poles of the 2^{nd}-order IIR filter used to identify the NDL system. The initial poles are located at the origin since the coefficients of the LMS filter are initialized to zero. As time progresses, the poles migrate towards the optimum solution. Since the filter is of length 2, the poles form a complex-conjugate pair. After "steady state" is reached, the poles oscillate about their respective positions in a manner that is indicative of a local linearization about an operating point.

Due to space limitations, we cannot demonstrate here the response of adaptive filters to various types of input signals. For system identification, the system is usually excited by band-limited white noise or chirp signals. This ensures that all the relevant frequencies of the system are excited and an appropriate dynamic model can be identified. These issues are discussed in detail in [23].

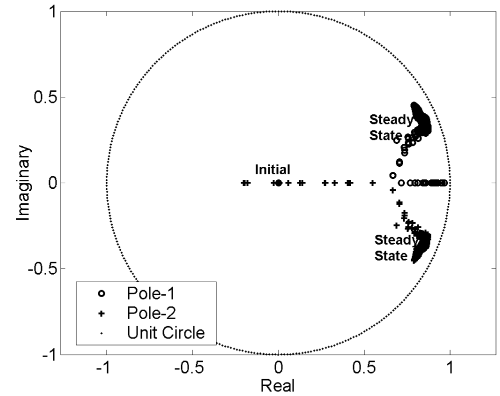


Fig. 16. Convergence history of the two poles of an IIR filter for the NDL system.

Conclusions

Linear and nonlinear adaptive filters relevant for the identification of SISO fluid-dynamic systems are presented in this paper. Recursive algorithms adaptively update the filter coefficients. In particular, LMS and RLS algorithms are summarized and the choice of model parameters is discussed. In the interest of space, only the LMS and RLS algorithms for a FIR model were presented in this paper. The algorithms for IIR and nonlinear filters are summarized in [23]. The real-time viability of the filters was tested by implementing them on a dSPACE DS1003 board that employs a TI TMS320C40 DSP processor. It was verified that the normalized LMS algorithm is significantly less expensive computationally than the RLS algorithm and also provided comparable convergence rates. In addition, the sampling frequency that can be achieved is sufficient to resolve frequencies on the order of several kHz prevalent in wind-tunnel tests. Faster DSP processors are available that will improve this performance and allow for simultaneous adaptive control.

The adaptive filters were applied to identify relevant dynamic systems – a nonlinear spring-mass-damper system and a nonlinear drag law. Low-order filters are able to identify the systems with good accuracy while maintaining stability. In addition, adaptive linear filters can identify nonlinear systems due to their time-varying nature. The FIR filter is inherently stable because of its structure but may lack the required accuracy. IIR filters are more accurate than FIR filters but additional computations are required to guarantee filter stability in real-time

applications. Finally, the nonlinear filters explored here are more accurate than the linear filters but are more expensive computationally. In addition, the design of a nonlinear controller based on a nonlinear adaptive identification scheme is a non-trivial task that needs to be addressed in real fluid-dynamic applications.

Future work will concentrate on evaluating these schemes with experimental data. The algorithms will be extended to multiple-input/multiple-output systems to study applications that require more than a single actuator and sensor. Finally, the system identification algorithms will be combined with an adaptive controller for real-time feedback control of cavity oscillations and airfoil separation.

Acknowledgements

This work was supported by a grant from AFOSR F49620-00-1-0284, monitored by Dr. Steven Walker. Additional support is acknowledged from NASA Langley Research Center grant NCC-1-01023, monitored by Dr. Michael Kegerise.

References

1. Cattafesta, L. N., III, Garg, S., Kegerise, M. A., Jones, G. S, "Experiments on Compressible Flow-Induced Cavity Oscillations," AIAA-98-2912, June 1998.
2. Williams, D. R., Fabris, D., Iwanski, K., Morrow, J., "Closed-Loop Control in Cavities with Unsteady Bleed Forcing," AIAA-2000-0470, 2000.
3. Rathnasingham, R. and Breuer, K. S., "System Identification and Active Control of a Turbulent Boundary Layer," AIAA 97-1793, June, 1997.
4. Mouyon, P., Casalis, G., Séraudie, A., and Prudhomme, S., "Feedback Control of the Laminar-Turbulent Transition Onset in a Boundary Layer by Suction", *Experimental Thermal and Fluid Science*, vol. 16, pp. 22-31, 1998.
5. Allan, B. G., Juang, J. N., Raney, D. L., Seifert, A., Pack, L. G., and Brown, D. E., "Closed-loop Separation Control Using Oscillatory Flow Excitation," NASA/CR-2000-210324, August, 2000.
6. Ljung, L., System Identification – Theory for the User, Prentice Hall, Upper Saddle River, NJ, 1987.
7. Widrow, B. and Stearns, S. D., Adaptive Signal Processing, Prentice Hall, Upper Saddle River, NJ, 1985.
8. Shynk, J. J, "Adaptive IIR Filtering," *IEEE ASSP Magazine,* Vol.6, Issue 2, pp. 4-21, April 1989.
9. Schetzen, M., The Volterra and Weiner Theory of Nonlinear Systems, Wiley, N.Y., 1980.
10. Bose, T., Joo, K. S., "The "Generalized Bilinear System": A Class of Nonlinear Systems," *ISCAS '94,* Vol.2, pp. 185-188, 1994.
11. Diaz, H. and Desrochers, A. A., "Modeling of Nonlinear Discrete-Time Systems From Input-Output Data," *Automatica,* Vol. 24, No.5, pp. 629-641, 1988.
12. Wiener, N., and Hopf, E. "On a class of singular integral equations," *Proc. Prussian Acad. Math-Phys. Ser.*, pp 696, 1931.
13. Haykin, S, Adaptive Filter Theory, Prentice Hall, Upper Saddle River, NJ, 1996.
14. Kuzminskiy, A. M., "A Robust Step-Size Adaptation Scheme for LMS Adaptive Filters," Proceedings DSP 97, Vol. 1, pp 33-36, 1997.
15. Feintuch, P. L., "An Adaptive Recursive LMS Filter," *Proceedings of IEEE*, Vol.64, pp. 1622-1624, November 1976.
16. Treichler, R. J., Johnson C. R., Larimore G. M., Theory and Design of Adaptive Filters, John Wiley and Sons, New York, 1987.
17. Netto, L. S., Diniz, P. S. R., Agathoklis, P., "Adaptive IIR Filtering Algorithms for System Identification: A General Framework," *IEEE Transactions on Education* Vol. 38, No.1, February 1995.
18. Coker, M. J. and Simkins, D. N., "A Nonlinear Adaptive Noise Canceller," *Proceedings of the 1980 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 470-473, 1980.
19. Mathews, V. J., "Adaptive Polynomial filters," *IEEE Signal Processing Magazine*, pp. 10-26, July 1991.
20. Nayfeh, A. H. and Mook, D. T., Nonlinear Oscillations, John Wiley & Sons, Inc., N.Y., pp. 162-195, 1995.
21. Rathnasingham, R., Breur, K. S., "Coupled Fluid-Structural Characteristics of Actuators for Flow Control," *AIAA Journal*, Vol. 35, No.5, May 1997.
22. Stonick, V. L., "Time-varying performance surfaces for adaptive IIR filters: geometric properties and implications for filter stability," IEEE Transactions on Signal Processing, Vol.43, Issue 1, pp. 29-42, January 1995.
23. Pillarisetti, A., "Real-Time Adaptive Identification of Dynamic Systems," M.S Thesis, Dept. of AeMES, University of Florida, July 2001.